INL/CON-06-01255
PREPRINT

Quantitative Risk Reduction Estimation Tool for Control Systems – Suggested Approach and Research Needs

International Workshop On Complex Network and Infrastructure Protection

Miles McQueen
Wayne Boyer
Mark Flynn
Sam Alessi

March 2006

The INL is a
U.S. Department of Energy
National Laboratory
operated by
Battelle Energy Alliance

QUANTITATIVE RISK REDUCTION ESTIMATION TOOL FOR CONTROL SYSTEMS
Suggested Approach and Research Needs

Miles McQueen, Wayne Boyer, Mark Flynn, Sam Alessi
Idaho National Laboratory[1]

Keywords: Risk Estimation, Control System Security, Network Security.

Abstract

For the past year we have applied a variety of risk assessment technologies to evaluate the risk to critical infrastructure from cyber attacks on control systems. More recently, we identified the need for a stand alone control system risk reduction estimation tool to provide owners and operators of control systems with a more useable, reliable, and credible method for managing the risks from cyber attack. Risk is defined as the probability of a successful attack times the value of the resulting loss, typically measured in lives and dollars. Qualitative and ad hoc techniques for measuring risk do not provide sufficient support for cost benefit analyses associated with cyber security mitigation actions. To address the need for better quantitative risk reduction models we surveyed previous quantitative risk assessment research; evaluated currently available tools; developed new quantitative techniques [17] [18]; implemented a prototype analysis tool to demonstrate how such a tool might be used; used the prototype to test a variety of underlying risk calculational engines (e.g. attack tree, attack graph); and identified technical and research needs. We concluded that significant gaps still exist and difficult research problems remain for quantitatively assessing the risk to control system components and networks, but that a useable quantitative risk reduction estimation tool is not beyond reach.

1.0 Background

The U.S. infrastructures that provide electricity, transportation, chemicals, foods, etc., are increasingly being networked to increase production and services, and simplify management. A control system usually serves as a key subsystem within these networks. Control systems are generally composed of sensors, controllers, computer systems, multiple communications channels, and human operators (see Figure 1).

1.1 Control System Connectivity and Standardization

For efficiency, control systems are increasingly being connected to the Internet and corporate business networks. In addition, control system sensors, controllers, operating systems, and console software are becoming more standardized and commercially available. These actions, although positive and generally efficient, open control systems to similar vulnerabilities and

[1] Idaho National Laboratory, 1955 Fremont St., Idaho Falls, Idaho, U.S.A.

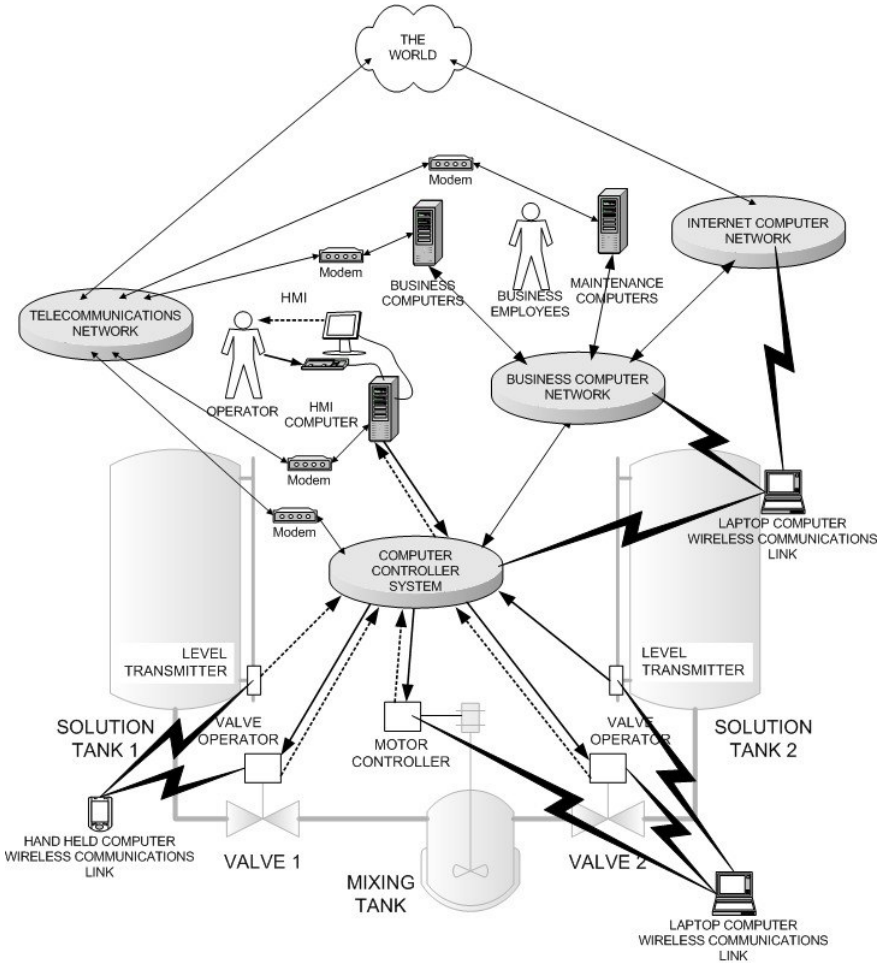

Figure 1 Simplified example of a control system applied to mixing two fluids remotely.

potentials for cyber attack that plague Internet-connected computers today. On average, it is estimated that seven new vulnerabilities in commercial software systems are identified every day [31]. Exposing the control systems used to manage our critical infrastructures to these vulnerabilities and threats are of great concern because once they are compromised an attacker can potentially assume control of the system, and drive it to a failed state causing damage and even loss of life [9].

To get some idea of the complexity level emerging from this increased connectivity and movement to standardized protocols and commercial-off-the shelf software, consider a typical set of software installed on three different computers. One high-level analysis model for the three hosts, one firewall, and one router contain on the order of 2^{90} potential system states, each of which might constitute vulnerabilities that allow attackers to control the critical assets of the system [28]. If the size of the system is increased to five hosts, the number of estimated states increases to 2^{229} states. Control systems in critical infrastructure (e.g. electrical grids) contain many more hosts and intelligent devices with potential vulnerabilities that increase the complexity to a point where a statement about the Internet made by a Notre Dame physicist, Dr. Barabási, rings true: "Increasingly, we are realizing that our lack of understanding of the Internet and the Web…is rooted in the absence of a scientific framework to characterize the topology of the network behind it." [1].

Therefore, attempts to increase the security of control systems must deal with this extreme complexity. The difficulty in analyzing and managing the associated risk is amplified by each

control system vendor offering unique system configurations that include different types of computer hardware and software operating system platforms. Some of these are legacy systems that date back 20-plus years. Users also have unique collections of sensors, communications media and equipment, and various generations of system components within their facilities. The movement is toward ever greater connectivity.

2.0 Risk Estimation

Problems, associated analyses, and, in particular, the risk model described in this paper may be dependent on the particular context being addressed. The context in which particular problems are found can vary by *purpose* (e.g., prevention, detection, ranking, etc.), *scale* (e.g., national, regional, sector, site, control system, component), *audience* (e.g., DHS, industry, academia), and *kind* (e.g., random process, intelligent game, etc.). Simply stated, a change of context demands a different abstraction of the problem. A problem, for instance, identified and applicable on a national level will require different thinking and solutions than those dealing with a local site issue. The rest of this paper focuses on the issue of evaluating the risk reduction to an individual site without consideration of the influence of other perhaps more easily attacked facilities.

2.1 Definition of Risk

One formal definition of risk is the probability of a negative event multiplied by the impact. Risk can thus be defined in this formal quantitative sense, as the probability of an unwanted consequence (j) occurring from an event (i) multiplied by the value of that consequence, or:

$$R_{ij} = P_i * C_j \tag{1}$$

In addition, the overall risk to the system being characterized, assuming independence of events and consequences, is defined as the sum of overall consequences and events, or:

$$R_T = \sum_i \sum_j R_{ij} \tag{2}$$

2.2 Proposed Probability Decomposition

The first term in the risk model (Equation 1) represents the probability of some event (P_i) occurring. This event, for the control system cyber security problem, is a successful attack that results in the consequence (C) being realized. Though simple to write down, this event probability can be difficult, and at times impossible, to estimate with any confidence.

A successful terrorist launched cyber attack (P_i) should be decomposed into multiple factors that correspond to the thoughts and actions of both attackers and defenders of a facility. The decomposition must match the problem space and should be reasonably intuitive to understand for both the risk model builder and the user. A proper decomposition allows model builders and users to clarify mental models, which enhances communication and supports proper interpretation of the model outputs.

A variety of decompositions of P_i have been suggested in the literature [2] [15] [22]. The decomposition we have chosen, which we will use as the general foundation of any risk model built for specific decisions, is specifically tailored for control systems, retains a focus on the intelligent adversary, and has intuitive meaning for the user. The equation for this decomposition is:

$$p_i = F(t, a, b, s, c) \tag{3}$$

where the terms on the right side are defined as "events" for facility i:

t = the facility is on the target list of terrorist groups

a = the terrorist group launches an attack against facility i

b = facility control system perimeter is breached

s = the internal system components necessary for a successful attack are compromised

c = a system event occurs that generates the consequence(s).

Equation 3 defines the occurrence probability of a control system mediated consequence as a function of variables related to circumventing system defense. The variables (Equation 3) are event related, but can be modeled and functionally related to final event probability in a number of different ways. For example, the variable t, could be modeled as the probability that a facility is on the target list or alternatively assumed to be 1.0 for a specific facility.

The general functional form of Equation 3 allows for various approaches to be used in estimating the probability of consequence. A variety of models can be constructed from the general form by making different assumptions about the terms. For example, if the terms are defined as predictor variables in, say, a general linear statistical model, estimates of coefficients are determined using information and data; hence, the estimate of probability of consequence. However, in this paper we will decompose the probability into a multiple of the conditional probabilities associated with an attacker achieving the "events" specified above. This decomposition is given as

$$P_i = P_t P_a P_b P_s P_c \tag{4}$$

3.0 Risk Reduction Estimation Tool

Our control systems risk reduction estimation tool prototype was constructed to demonstrate the overall concept to customers, obtain additional requirements, study the feasibility of quantitative risk reduction estimation, and more firmly define research and development needs. The prototype integrates security requirements—both technical and administrative controls, a network description with known vulnerabilities, and a variety of underlying risk reduction estimation models. Other products, tools, and functions were discussed but are not currently part of the prototype[2]. Overall, the prototype presents the user with an interface that shows risk reduction as various security measures are added or removed from the network, see Figure 2.

The primary elements of the interface are the threat level, possible defensive mitigation measures, complete control system network specification, a time to compromise metric to assist in risk reduction estimation, an estimate of the potential consequences to the system owner, and percent risk reduction estimation as defensive measures are added or removed. Each of these is briefly described and discussed in the rest of this paper.

3.1 Threat Level

By threat we mean an intelligent adversary who may choose to attack the infrastructure through its control system. A threat must have *intent*, *capability*, and *opportunity*. Intent implies that the adversary values the damage that might be caused if they are able to successfully attack the infrastructure more than the cost they might have to bear. Intent affects the probability that the

[2] Examples of other tools that could be integrated include, user requirements traceability, alternate network views including hierarchical, geospatial, time to compromise, etc. and integration with CERT incident management systems.

infrastructure will be targeted and attacked. Capability relates to the conditional probability that the adversary will actually attack and be successful in causing damage to the system. And opportunity means that the adversary must have access to the vulnerabilities in the system so that his capabilities may be used to cause the desired damage.

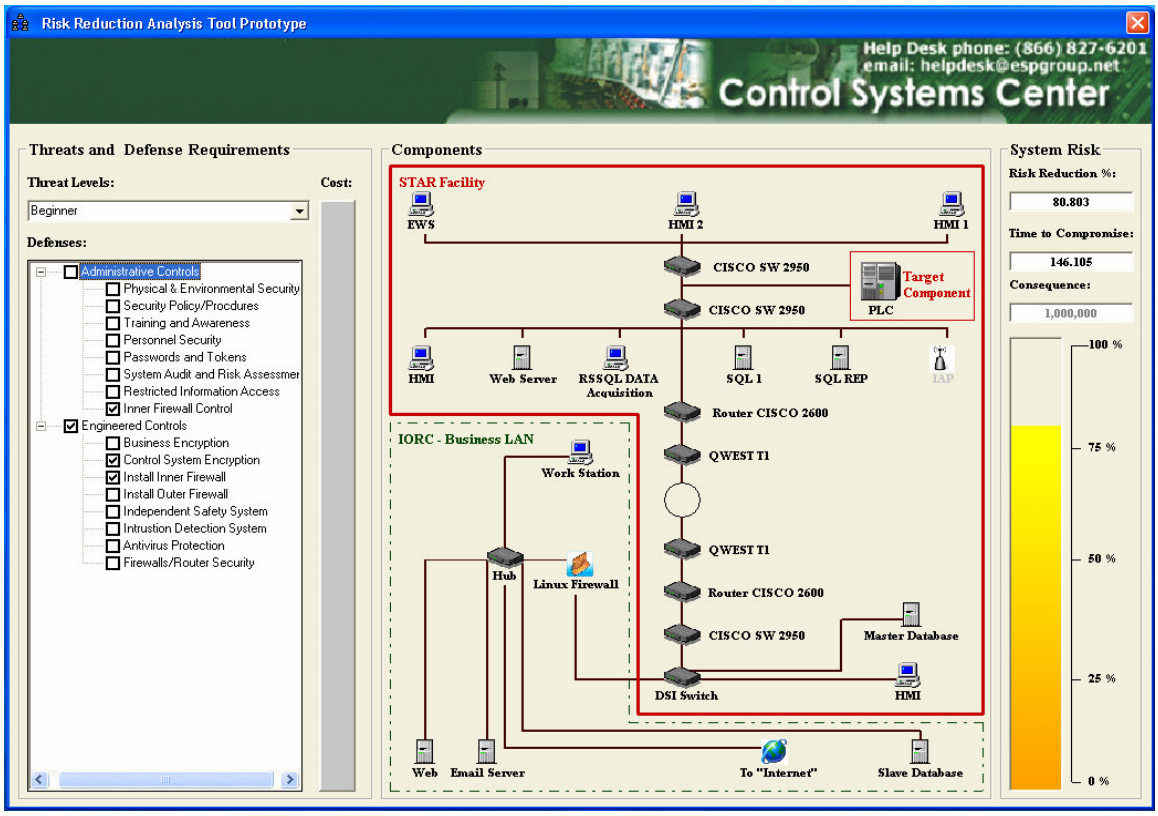

Figure 2. Risk reduction estimation prototype interface.

A variety of taxonomies have been used to partition threats into sets that are useful for analysis. Unfortunately, each of the taxonomies that we found mixed the concepts of intent, capability, and opportunity in such a way as to make a categorization of a specific group problematic. For example, using the Blackhat taxonomy as shown in Table 1 how would one categorize an insider with limited hacking skills who is a member of an Islamic terrorist cell? That insider belongs to at least three of the categories in the taxonomy. Each of the taxonomies in Table 1 have a similar weakness in their lack of clarity.

We propose that threat taxonomies for risk estimation be thought of as three dimensional objects with dimensions consisting of intent, capability, and opportunity:

$\text{Threat}_i = (\text{Intent}_i , \text{Capability}_i , \text{Opportunity}_i)$.

Further, each of these dimensions should be explicitly discussed and preferably defined before a risk assessment is performed. An initial taxonomy that we propose consists of two types of opportunity; four types of capability; and six categories of intent.

3.1.1 Threat Intent

Intent is probably the most difficult threat dimension to firmly establish. The initial categorization that we used partitioned intent into the categories of rock star, vandal, criminal

enterprise, foreign intelligence, terrorists, and nation states. A rock star is an individual or group whose sole intent is to gain notoriety by attacking infrastructure. For a rock star, damage is not a particularly desired outcome unless it enhances their renown as skilled hackers. A vandal represents a threat intent on doing damage for damage sake by cyber attack. A criminal enterprise includes attacks by organized crime pursuing monetary profit, and corporate spying in pursuit of information that would provide competitive advantage. Foreign intelligence includes all government sponsored entities whose mission is to acquire information that can be used to advance their countries interests. Terrorist intent is to generate infrastructure events to instill fear in a civilian population to further their political aims. A terrorist, by our definition, is not controlled by a nation state. A nation state represents a threat intent on damaging another countries infrastructure in order to further their nation's strategic goals. Other categories related to intent may be needed as the threat landscape continues to evolve.

Table 1 Three threat taxonomies.

Cyber Threats to Critical Infrastructure (blackhat) [25]	**Threats to Critical Infrastructure** [9]	**Cyber Attacker Categories** [5]
Hacker/Script Kiddies/Hobbyist	Criminal group	Terrorist groups
Disgruntled Employee	Foreign intelligence services	Nation States
Insider aiding others	Hackers	Anti-Capitalism/Anti-Globalization and terrorist Sympathizers
Hacktivist	Hacktivists	Thrill Seekers
Industrial Espionage	Information Warfare	
Foreign Espionage	Insider threat	
Terrorist	Virus writers	
State Sponsored Attack		

3.1.2 Threat Capability

Capability as it relates to the estimation of risk to the infrastructure from attacks through the control system should focus on the adversary's cyber attack skill. This might involve their technical skills in finding and exploiting electronic system vulnerabilities or their skills in social engineering. The initial partitioning we created relates strictly to technical skill and includes the categories of novice, beginner, intermediate, and advanced hacker.

3.1.3 Threat Opportunity

Because of its ease of use and categorization we partitioned **opportunity** into the rough categories of insider and outsider. By definition an insider is defined as someone who has approved user privileges to one or more network devices that are inside or part of the perimeter of the control system under evaluation. This definition aids in understanding that the insider my not only initiate attacks from inside the systems boundary but may also just prepare the system for external attack (e.g. the insider may install a back door for a remote compatriot).

3.1.4 Percent Risk Reduction Metric

As previously mentioned in Equation 4, we have decomposed the probability of generating a consequence from an attack on the control system into the probability of being targeted, and the conditional probabilities of being attacked, of the attack breaching the systems perimeter defenses, of the necessary internal system components being compromised, and of the desired consequence being generated. A more complete discussion of each of these probabilities can be found in [12].

These probabilities may be different for each category of attacker. Since our concerns revolve around terrorist groups and nation states we had to address the issue of infrastructure owners not having much if any information relative to those groups' specific targets or their capabilities for

cyber attacks into the infrastructure control systems. Further, while some control system incident data has been collected [11], the data is still too sparse to yield any confidence in the underlying statistics. Consequently, estimating the probability of attack is currently unresolved.

For these reasons we were not confident that an absolute risk metric was currently feasible so we decided on a different measure than absolute risk. By estimating the percent risk reduction, and assuming that the individual probabilities are independent, the unknown probabilities can be ignored since we can assume they stay unchanged as defensive measures are added or removed. In Equation 5, the probabilities with subscripts labeled (new) are those that have changed due to changes in system defensive controls or configuration.

$$\%\text{RiskReduction} = 100 * \left[1 - \frac{P_t P_a P_{b(new)} P_{s(new)} P_{c(new)}}{P_t P_a P_b P_s P_c} \right] \quad (5)$$

3.1.5 Threat Specification In Risk Reduction Estimation Prototype Tool

In our Risk Reduction Estimation prototype we allow some indirect indication of threat intent by allowing the user to specify which nodes on the system are targets of the threat.

In our prototype, the user can choose between four threat levels: Novice, Beginner, Intermediate, and Expert. These threat levels represent the four capability levels we postulated in previous risk analysis research [17] and chose to use in the prototype tool. These models and their associated measures of time to compromise, rate of attack, etc… present significant investment opportunities for research development and validation.

In the Risk Reduction Estimation prototype tool the percent risk reduction values and time to compromise measures are dependent upon the threat level chosen by the user. A threat level must be chosen by the user before a defense list is shown and available. The four threat levels chosen for the prototype have specific definitions that relate to their individual skill levels. The definitions may be found in [17]

3.2 Defensive Mitigation Measures

A control system usually includes a control room environment which may consist of standard computers running pervasive operating systems such as Windows or Solaris. The applications on these computers are usually control system specific unless implementing a standard functional need such as a data base. Attached to the control room computers is a network of communication and control devices such as remote terminal units and programmable logic controllers. The protocols and applications running on these devices are vendor dependent and often installation selectable. Often the control system specific components are older software and hardware than typically found on standard enterprise networks or even at home. More and more frequently the control network is being connected to the enterprise network. Further, this interconnection between the enterprise network, the control room(s), and one or more control system subnets using a wide range of technology leads to a puzzling security picture. This creates an even greater need for a well thought out, specified, and implemented security program whose posture can be clearly and definitively expressed to a variety of management levels through security and risk metrics related to system components and defensive mechanisms.

Defensive mitigation measures include both administrative and technical controls. The administrative controls include items such as a written security policy, adoption of best practices, specification and collection of security metrics for each level of reporting, security maintenance programs, regular vulnerability assessment audits, and security education and training for those

responsible for day to day operations. The technical controls include tangible defensive components such as a firewalls and intrusion prevention systems, adoption of appropriate cryptographic protocols to prevent replay or man in the middle attacks, network architectures to optimize security within operational constraints, and technical security metric collection and evaluation.

3.2.1 Administrative Controls

Administrative controls must include measures that are well accepted within the enterprise security management community and other mitigation measures tailored for the unique needs of control systems. An assessment of security for a given facility should be based on codes, standards, and adopted policy. One example of this approach may be found in [13] where over 1000 security survey questions were culled from ISO 17799 *Code of Practice for Information Security Management*, NIST SP 800-26 *Security self-Assessment Guide for Information Technology Systems*, and the OCTAVE *Catalog of Practices*. An example of control system specific questions is a survey currently under development at INL based on the NIST SPP-ICS baseline set of security requirements for new control systems.

3.2.2 Technical controls

Technical controls should be based on best practices but also tailored for each individual control system. The adoption and configuration of security devices and mechanisms would ideally be based on the cost of installation and up keep of the device as compared to the risk reduction it provides. Unfortunately, the capability for quantitatively measuring the effectiveness of security mechanisms is currently beyond the state of the art. However, recent research efforts are beginning to bear some fruit and technical quantitative security measures appear to be a promising area for research and development over the next five years. Also, as previously mentioned, it is an unresolved issue of how to calculate the absolute risk in dollars (or lives) to a system. This indicates an opportunity for improved attack detection and assessment research, and nation or world wide attack and event data collection effort.

3.2.3 Security Controls In The Risk Reduction Estimation Prototype Tool

In the Risk Reduction Estimation Prototype tool, the defense list indicates which controls are available to reduce the risk of a successful attack on the system. The defenses in this list are related to both administrative and technical system defensive measures as previously discussed. The defenses chosen by the user and applied to the system are used to calculate percent risk reduction values. The defensive measure list requires more investigation. At too high a level of abstraction the value of the control is so generic that meaningful information about the reduction in risk is difficult to assess. But too low a level of abstraction (e.g. the actual firewall rule set) will provide too much data and might well be difficult to convert to a common, higher level, security abstraction for each vendor or component. Research into the appropriate levels of abstraction for each generic device and into tools that would take specific device data and transform them into useable standard form and semantics is still needed.

3.3 Control System Network Specification

The control system network specification contains a visual display of the system of interest, its components, and their connections. Information concerning the software and associated vulnerabilities related to a specific component should be available by clicking on the component of interest. Also, as specific hardware related defenses are implemented, such as a firewall installation, the component must be added to the network specification. One need here is for a passive discovery tool that may be connected to a control system, perhaps in a variety of locations, that builds a model of the system through analysis of the network traffic. A useable discovery tool requires some applied research and a significant amount of developmental effort.

One recent research initiative in this area is [6]. Another need is for a very small and safe host based software vulnerability scanner that assesses the potential flaws in component configuration, identifies known vulnerabilities, and estimates the level of unknown risk. Both the network discovery tool and host based scanner will most likely be independent of the risk reduction estimation tool but they are necessary to provide critical topological and vulnerability information about the control system.

3.4 Security Models and Measures

For at least three decades security measures and models have been investigated. The results of these investigations have been mixed. In the subfield of cryptography there has been significant success in both designing cryptographic ciphers and protocols, and analyzing them to establish correct behavior. The analysis has generally been through open expert review [20], formal methods [7], and mathematical and algorithmic analysis (e.g. the difficulty in factoring) [8]. Making certain assumptions explicit such as *the cipher is known but the key is secret* has helped lead to a common framework in which to analyze the security of ciphers and protocols. Generally the analysis leads to either an estimate of average time to discover the key or to an estimate of the number of tries that would be needed. Passwords are another area that is both well studied and well understood.

Passwords may be categorized by their length and by their entropy (roughly speaking entropy is a measure of the randomness in the characters). Greater length and greater entropy lead to increasing the average number of tries and time necessary for discovering the password. The practical weaknesses have also been well studied including the fact that most users still select passwords that have very low entropy. Administrative controls have been made available to help force users to make password selections that are more difficult to break.

It is worth noting that due to the solid scientific underpinning of the analysis, the explicit statement and acceptance of the assumptions (e.g. both plain text and associated cipher text are available), and the intuitive and meaningful measures used, the security provided by cryptographic ciphers, protocols, and passwords are well understood. Unfortunately, the success in developing measures and models in other security subfields have been rather less successful. In the measures and models we have looked to use for analyzing the risk reduction in a control system from the adoption of defensive measures there is little consensus on measures or models.

3.4.1 Security Models

Security models have been developed for many scales of analysis. On a national scale, attempts are being made to model the interdependencies of critical infrastructure through simulations [19]. Other, game theoretic, approaches for estimating the risk to a given facility by including the value and defense posture of other infrastructure components have been proposed by [15] and [10]. For a given facility risk assessment techniques have included surveys based on standards and codes [30], and models that don't need explicit knowledge of the system connectedness or vulnerabilities [23]. For the Risk Reduction Estimation tool we were interested in a model that would make use of all the information that might be available about the control system components, communication channels, and protective measures.

Many models have been proposed for assessing the security posture of an electronic network based on detailed data about the system being evaluated. The more recent attempts have included different types of fault trees, attack trees, attack graphs, and game theoretic models. To date we have applied the first three types of models to our calculation engine used in our risk reduction estimation. Each has their strengths and limitations which are discussed below. The fourth set of

models, making use of game theoretic principles, seems quite promising but we have not made use of them yet.

3.4.1.1 Fault Trees

A good review of fault trees may be found in [24]. Fault trees have been applied successfully too many reliability problems over the last few decades. They have been useful in a variety of contexts. The system detail they capture is dependent on the risk associated with failure and the time available for analysis. In our application of the method to detailed risk assessment in a control system three significant difficulties arose.

The first difficulty was in assessing the failure rates for the different nodes and communication channels in a control system. Since the threat is intelligent and adaptive it is not at all obvious how to estimate the failure rates since their randomness is called into question. We refer to successful attack of a component as a compromise in order to emphasize the lack of a random failure process.

The second difficulty lay in the dense interdependencies between component failures. For example, suppose a control room is made up of seven computers with one of them being an intermediate target. Then an external attacker may be able to reach the target directly; or the attacker might have to first compromise one of the other machines before compromising the intermediate target, or the attacker might prefer to compromise one of the other machines first in order to compromise a second machine with which there are trust relationships followed by a compromise of the intermediate target; etc. Without credible component compromise rates and with the large number of compromise paths just suggested it is not possible to winnow the fault tree down to just those failure modes that are the most likely.

The last difficulty we incurred in our application of a fault tree was the need to automate the generation of the underlying fault tree based on the network topology and the technical defensive measures that are added, removed, or modified.

We brought in INL experts to aid us in applying fault tree technology since the methodology is so intuitive and well understood. Unfortunately, while initially optimistic, our experts were unable to overcome the significant obstacles mentioned above. Consequently while we believe that fault trees may have a role to play at a higher level of abstraction, they do not seem to be a good solution when faced with detailed control system information and an intelligent adversary.

3.4.1.2 Attack Trees

Attack Trees for security analysis were first proposed in 1999 by Schneier [27]. The purpose in constructing an attack tree is to aid a security analyst in identifying the most likely attack paths, and to help guide effective application of limited defensive resources. Since attack trees are not comprehensive in their identification of all possible attack sequences, they are highly dependent on the experience of the team of security analysts who build them. Another problem in applying this technique is that the process of building an attack tree does not appear to be amenable to automated construction. A final problem is that the weights needed for each leaf node in order to discover the best application of defensive resources is dependent, once again, on expert elicitation.

Despite these drawbacks we are pursuing collaborative work in building a generic control system attack tree, and hope that, in the short term, it will be a useful aid for security analysts in identifying reasonable applications of defensive resources in control systems. Figure 3 is an

example attack tree that resulted from this effort. We do not believe that attack trees constitute a long run solution for detailed security and risk assessment.

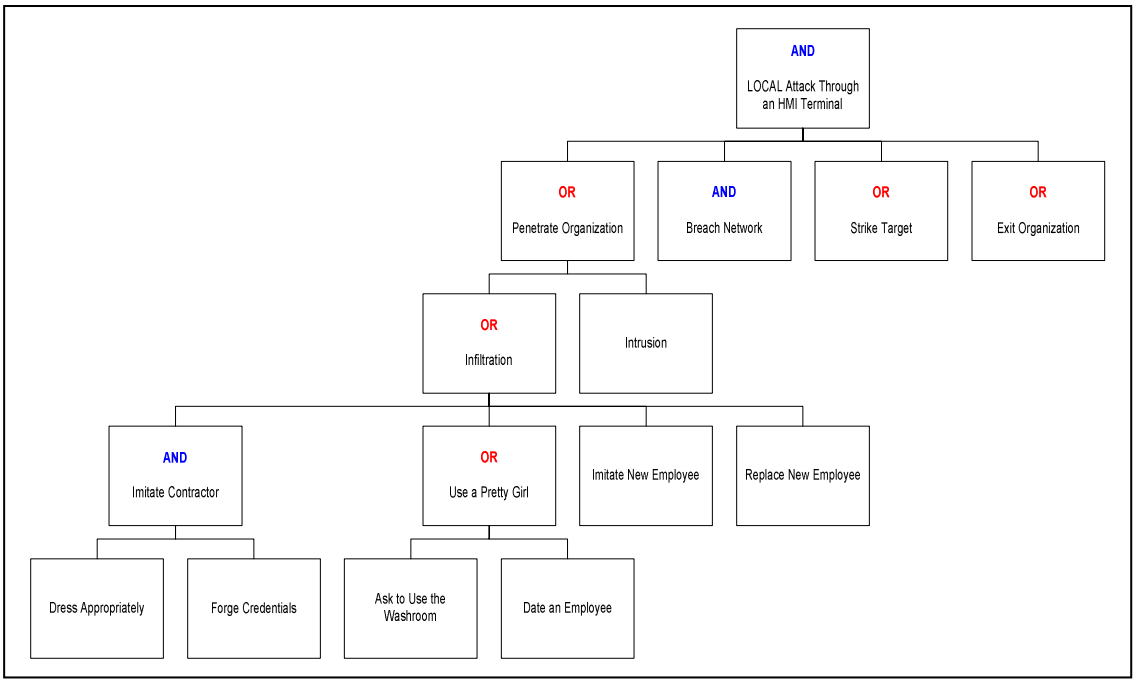

Figure 3 Example partial attack tree. Local attack through an HMI terminal.

3.4.1.3 Attack Graphs

In principle attack graphs generate all attack paths available to an attacker [29]. Many different algorithms have been proposed for automated generation of attack graphs and they all depend on fairly detailed knowledge about the networks topology and the vulnerabilities on each node and communication channel. We encountered three issues in the use of attack graphs when we applied them to our test network. The first is that the runtime of the model checking algorithms suffer from exponential growth and currently become impractical for use in modeling even very small networks. The second problem is that since the number of attack paths grows exponentially in the number of nodes, the graph quickly becomes unwieldy. Consequently, the graph is currently only suitable for automated analysis. And thirdly, what security and risk analysis conclusions can be drawn from the graphs are unclear.

When we applied the model checking attack graph generation methods proposed by Sheyner et. al. [29] to our test network, we were not surprised to find that they suffered from exponential growth in runtime relative to the number of nodes in the network. What did startle us was that the exponential growth was so bad that even a small subset of nodes in our test network quickly overwhelmed the model checker. The poor runtime performance of the attack graph toolkit can be seen in the data we collected for simulated networks of various sizes and graphically presented in Figure 4.

With slight modifications to the underlying attack assumptions (e.g. monotonicity) some quite clever polynomial growth algorithms have been discovered and applied. One of these recently invented techniques has been instantiated in a tool suite named MulVal whose details may be found in [21]. We applied this research tool to the same set of simulated networks as before and

the vastly superior runtime performance of MulVal is clearly shown in Figure 4. We are currently collaborating with the originators of MulVal to further research and development into the entire tool suite for application to control system risk analysis.

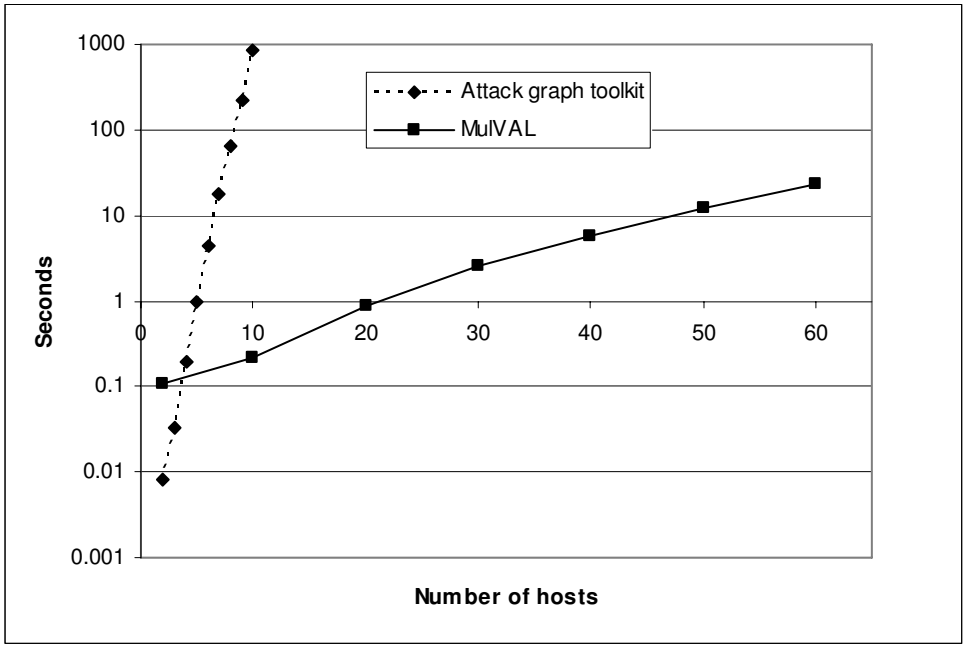

Figure 4. Graph generation time of Attack Graph Toolkit and MulVal (plotted on a log scale) for simulated networks of various sizes and 5 vulnerabilities per host.

The issue of the number of attack paths growing exponentially in the number of nodes is to be expected. There are a number of ideas that have been proposed to make the graph more useable by risk and security analysts. These involve ideas for pruning the graph, only displaying user designated portions of the graph, providing a query system that determines what portion of the graph to display based on the question being asked by the analyst, or developing an automated process to provide a meaningful, higher level abstraction, of the graph. Another technique and one we are currently investigating, is simply to use the attack graph as input to an underlying analysis engine and never display the graph to the analyst.

The last issue is a question of what analysis can be done with the graph once it has been generated. Unfortunately, the current state of the art does not provide any scientifically based mechanisms for weighting the attack graph edges. Thus we are currently left with little else than applying basic and essentially unbelievable measures and interpretation of the attack graphs. This issue is discussed below in section 3.4.2 Security Measures.

3.4.1.4 Game Theoretic Models

Game theoretic models applied to cyber security have recently been investigated [3] [14]. The strength of these approaches is the rigorous mathematical modeling of intelligent adversarial relationships. The approaches are not fully developed and it will be interesting to see the results of future research. A current weakness in each model we have seen is a lack of guidance in how to generate the parameters necessary to make the models useable. (This is also true for all other models we investigated).

3.4.2 Security Measures

Similar to security models, security measures have been proposed and evaluated for many scales of analysis. Few have gained acceptance within the communities they were aimed at. For the level of analysis we are attempting with the Risk Reduction Estimation tool one can find models that require many different security measures for individual functional components including *difficulty*, *effort*, *time*, *rate of vulnerability discovery*, *patch rate*, and *probability of component compromise*. Each of these have some merit including the ill defined measures of difficulty and effort.

Unfortunately, little guidance is provided in the literature for how one might collect or estimate the most important of these measures. The reason for this is that it is difficult and there is no accepted scientific or engineering process for determining them. For example, while known vulnerability counts can be made for each component it is simply unknown how to convert that information to a probability of component compromise. One recent effort to at least approach the issue can be found in [26].

The issue of estimating unknown vulnerabilities also remains an outstanding research issue with little research done to date. We are pursuing collaborative research in attack surfaces [16] and time-to-compromise estimations in an attempt to stimulate research into statistical estimates for relevant measures. We believe that significant opportunities exist for improving the state of the art in this domain.

One other area where reasonable security measures are sorely lacking is in defensive devices. For example, recent work by [4] indicates that simple obfuscations of well known viruses may easily bypass anti-virus software. Worse, the specific obfuscations needed can be reverse engineered from the virus signature files. Consequently, what measures should be used in assessing the security provided by anti-virus tools is more of an open question than the associated vendors would seem willing to indicate. This situation applies to intrusion detection and protection devices and many of the other defensive mechanisms available for installation in a control system environment. We are now investigating possible measures for a few of these devices.

For our prototype tool we developed a process for estimating the time it would take an attacker to successfully attack an individual component of the system. This value is dependent upon the threat level and defenses chosen by the user. This is just one of many metrics that might be reasonably applied as an intermediate measure to aid the estimation of risk reduction. Whatever metrics are chosen, they must be intuitively associated with risk and defensibly mapped to risk reduction. Some of the measures and metrics must represent some estimate of unknown vulnerabilities and the susceptibility of the component to attack because of the unknowns. As mentioned earlier, some recent work in this area includes attack surface estimation. A significant research effort is still required to determine useable security measures and metrics for software components.

3.5 Consequence Value

Understanding the possible range of consequences from a successful cyber attack on an infrastructure's control system is necessary for risk estimation. In more traditional areas of risk analysis (e.g. nuclear power), estimating consequences is a very difficult task that typically requires a significant amount of analysis and site-specific data about the facility, as well as the co-located facilities and populations. These detailed analyses can take man-years of effort, even with complete data availability. Consequently, this approach is usually only applied to high-consequence, high-cost operations.

The introduction of cyber attacks only complicates this already difficult analysis. The types of additional complications that can arise include: the likelihood of events that may be missed in the analysis due to the tremendous variability in the types of attacks; a dramatic difference in attack consequence due to the level of detailed knowledge the attacker has about the control system; and the difficulty of making accurate predictions from historical events since there have been so few actual consequences from cyber attacks on control systems that have been reported.

Some of the existing validated consequence codes (e.g. air dispersion, toxicological, and economic techniques) used in such applications as geographic information systems (GIS), probabilistic risk analysis (PRA), and human reliability analysis, might be adapted or refined for use in risk estimates from attacks on control systems used in critical infrastructure.

In the current Risk Reduction Estimation tool, all of the above complications have been removed and the consequence value is simply determined by the user and indicates a dollar cost associated with a successful attack on the system. In general one would suppose that as the control system changes and defensive mechanisms are added and deleted then the possible consequences might change as well. However, that issue requires further research and development activities.

3.6 Percent Risk Reduction Value

The percent risk reduction value is provided by the underlying quantitative risk calculation engine and is determined based on the threat level and defenses chosen by the user. The percent risk reduction scale bar is directly related to the percent risk reduction value provided by the underlying quantitative risk calculation engine. As discussed previously, the underlying security models and measures is one of the most important and difficult areas requiring significant additional research efforts and insights.

4.0 Conclusion

The overarching goal is to make the creation of a quantitative risk reduction estimation tool feasible and generalizable to any control system environment. We demonstrated the need for significant additional research and development efforts across a number of disciplines, and identified specific technological barriers that need to be overcome before a complete risk reduction estimation tool can be produced. Increased research emphasis in assessing the unknown risks in software components, establishing some baseline quantitative security measures and metrics, and creating more practical risk reduction calculational engines is needed. Economic consequence estimation models, particularly for ripple effects across the economy, also need significant research. However, even with these formidable research and development challenges we believe that the value of a quantitative risk reduction estimation tool is significant, may shortly be able to provide improved estimations over current qualitative techniques, and should be pursued.

References

[1] Barabási, Albert-László (2001). *The Physics of the Web. Physics World*, July 2001.
[2] Beitel, G. A., D. I. Gertman, and M. M. Plum (2004). Balanced Scorecard Method for Predicting the Probability of a Terrorist Attack, *Presented at Risk Analysis 2004,* September 27–29, 2004*,* Rhodes, Greece.
[3] Bier, V. M., A. Nagaraj, and V. Abhichandani (2005). Optimal Allocation of Resources for Defense of Simple Series and Parallel Systems from Determined Adversaries, *Reliability Engineering and System Safety,* 87:313-323.

[4] Christodorescu, M. and S. Jha (2004). Testing Malware Detectors, Proceedings of the International Symposium on Software Testing and Analysis (ISSTA'04), July 11-14, 2004, Boston, Massachusetts, USA.
[5] Cortes, W. I. (2004). Cyber Terrorism Post 9/11 In The Western Hemisphere, *Monograph presented to the Inter American Defense College*, April 2004, Washington, DC, USA.
[6] De Montigny-Leboeuf, Annie, Frédéric Massicotte (2004). Passive Network Discovery for Real Time Situation Awareness, *RTO IST Symposium on Adaptive Defense in Unclassified Networks*,Toulouse, France, 19-20.
[7] Donovan, Ben, Paul Norris and Gavin Lowe (1999). Analyzing a Library of Security Protocols using Casper and FDR, *Proceedings of the Workshop on Formal Methods and Security Protocols*.
[8] Fahn, P. and M.J.B. Robshaw (1995). Results from the RSA Factoring Challenge. Technical Report TR-501, version 1.3, RSA Laboratories, January 1995.
[9] GAO (2004). *Critical Infrastructure Protection – Challenges and Efforts to Secure Control Systems*, GAO-04-354, U.S. General Accounting Office, March 2004, http://www.gao.gov/.
[10] Hausken, K. (2002). Probabilistic risk analysis and game theory, *Risk Analysis*, 22, No.1, (2002), pp.17-27.
[11] INL/EXT-05-00392 Revision 0 (2005). Industrial Security Incident Database (ISID) *Idaho National Laboratory report prepared for U.S. Department of Homeland Security*, June 6, 2005.
[12] INL/EXT-05-02585 Revision 1 (2005). Control Systems Risk Decision Methodology, *Idaho National Laboratory report prepared for U.S. Department of Homeland Security*, March 23, 2005.
[13] Johansson, E., M. Ekstedt, P. Johnson (2006). Assessment of Enterprise Information Security – The Importance of Information Search Cost, *Proceedings of the 39th Hawaii International Conference on System Science*, Kauai, Hawaii, USA.
[14] Lye, K.W. and J.M. Wing (2005). Game Strategies in Network Security, *International Journal of Information Security*, .
[15] Major, J. (2002). Advanced Techniques for Modeling Terrorism Risk, *Journal of Risk Finance*.
[16] Manadhata, P., J. Wing (2005). An Attack Surface Metric, CMU-CS-05-155, *School of Computer Science*, Carnegie Mellon University.
[17] McQueen, M.A., W. F. Boyer, M. A. Flynn, G. A. Beitel (2005). Time-to-compromise Model for Cyber Risk Reduction Estimation, *First Workshop on Quality of Protection*, Milan, Italy.
[18] McQueen, M.A., W. F. Boyer, M. A. Flynn, G. A. Beitel (2006). Quantitative Cyber Risk Reduction Estimation Methodology for a Small SCADA Control System", *Proceedings of the 39th Hawaii International Conference on System Science*, Kauai, Hawaii, USA.
[19] NISAC, National Insfrastructure Simulation and Analysis Center, http://www.lanl.gov/orgs/d/nisac/pdfFiles/nisac.pdf
[20] NIST Global Information Security Competition (2000). http://www.nist.gov/public_affairs/releases/g00-176.htm
[21] Ou, X., S. Govindavajhala, A. W. Appel (2005). MulVAL: A Logic-based Network Security Analyzer, *14th Usenix Security Symposium.*
[22] Rinaldi, S. (2004). Modeling and Simulating Critical Infrastructures and Their Interdependencies, *Proceedings of the 37th Hawaii International Conference on System Science*, Hawaii, USA.
[23] RiskWatch, http://www.riskwatch.com/isa.asp.

[24] Roberts, N. H., V. W. Vesely, D. F. Haasl, and F. F. Goldberg (1981). Fault Tree Handbook, *Systems and Reliability Research, Office of Nuclear Regulatory Research. U.S. Nuclear Regulatory Commission*. Washington, D.C. 20555.
[25] Sachs, M. H., Parker, T., Miller, T. (2003). Adversary Characterization and Scoring System, *Presentation at Black Hat*, July 28-31, 2003, Las Vegas, Nevada.
[26] Schiffman, M., A. Wright, D. Ahmad and G. Eschelbeck, (2004). The Common Vulnerability Scoring System, *National Infrastructure Advisory Council, Vulnerability Disclosure Working Group, Vulnerability Scoring Subgroup*, CVSS Evaluation Draft 1.0, July 27.
[27] Schneier, Bruce (2000). Attack Trees, *Secrets and Lies*, pp. 318-333, ISBN 0-471-45380-3.
[28] Sheyner, O., and Jeannette Wing, (2004). Tools for Generating and Analyzing Attack Graphs, *Carnegie Mellon University*.
[29] Sheyner, O., J. Haines, S. Jha, R. Lippmann, and J. M. Wing (2002). Automated Generation and Analysis of Attack Graphs, *Proceedings of the IEEE Computer Society Symposium on Research in Security and Privacy*, Berkeley, California, pp. 273–284.
[30] Swanson, Marianne (2001). NIST Special Publication 800-26 Security Self-Assessment Guide for Information Technology Systems, November 2001.
[31] Symantec News Release (2004).
http://www.symantec.com/press/2004/n040920b.html

Author Biographies

Miles McQueen is an Advisory Engineer at the Idaho National Laboratory and Computer Science Faculty, University of Idaho, Idaho Falls. Miles has over twenty-five years of complex system level analysis and design including real-time systems, sensors, simulations, and security. He has over fifteen peer-reviewed publications in the area of survivable systems and computer security. He has done advanced work in Computer Science and Economics where his emphasis was in micro-economics. Miles currently teaches class in Analysis of Algorithms, Computer Security, and Software Engineering Metrics. Miles holds degrees in Computer Science, Mathematics, and Economics.

Wayne F. Boyer, Ph.D. is an Advisory Engineer/Scientist at the Idaho National Laboratory (INL) in Idaho Falls, Idaho and a Computer Science instructor for University of Idaho, Idaho Falls. Before joining INL he was a member of technical staff and technical supervisor at AT&T Bell Laboratories in Whippany, New Jersey and Denver, Colorado. His current research interests are in parallel computing and network security for distributed control systems. Wayne holds degrees in Electrical Engineering and Computer Science.

Mark Flynn has worked at the Idaho National Laboratory (INL) since October 1999. During that time, he has been involved in various projects geared to ensuring the safety of INL computer systems. He is currently a member of a team charged with the research, design, and development of offensive hacking tools, intrusion detection systems, and vulnerability scanners.

Sam Alessi, Ph.D. is a Sr. Advisory Engineer and Director of Systems Program, U of I, Idaho Falls. Dr. Alessi has over 17 years experience is systems and software design including associated organizational development and assessment. An area of specialty involves human inquiry approaches for software requirements and public involvement.